

# /etc/hosts

- Objet : hosts
- Niveau requis : [avisé](#)
- Commentaires : *Configuration du fichier /etc/hosts.*
- Débutant, à savoir : [Utiliser GNU/Linux en ligne de commande, tout commence là !](#) 😊
- Suivi : [à-placer](#)
  - Création par [smolski](#) 04/01/2015 avec l'aide de [raleur](#) chez qui j'ai tout pompé ! 😊
  - Testé par <...> le <...> [Fix Me!](#)
- Commentaires sur le forum : [Lien vers le forum concernant ce tuto](#) <sup>1)</sup>

## Nota :

Contributeurs, les [Fix Me!](#) sont là pour vous aider, supprimez-les une fois le problème corrigé ou le champ rempli !

## Comment ça marche ?

Une mise au point pour commencer : /etc/hosts concerne la résolution locale de noms d'hôtes alors que /etc/apache2/sites-enabled concerne les sites gérés par apache (virtualhosts). Ce sont deux domaines qui n'ont strictement rien à voir.

- La résolution de noms ne se préoccupe pas des sites d'apache,
- apache se ne se préoccupe de la résolution de noms pour gérer les sites.



Il n'y a que pour le bon fonctionnement d'un client (navigateur web) que les deux doivent correspondre.

Voici ce que fait un navigateur pour accéder à **http://site/ressource** :

1. **Résolution** du nom d'hôte "*site*" pour déterminer son adresse IP.
2. **Connexion** TCP à cette adresse sur le port 80.
3. **Envoi** d'une requête HTTP pour la ressource "*/ressource*" sur le site "*site*".

On voit que pour que tout cela fonctionne, le serveur web qui répond à l'adresse IP obtenue par la résolution du nom "*site*" doit héberger un site nommé "*site*". L'adresse IP locale réelle du serveur peut être totalement différente, peu importe.

## Explications

Revenons maintenant au fichier /etc/hosts.  
Sa syntaxe est décrite dans sa page de manuel :

## man hosts

Il y est bien précisé qu'il ne doit en principe y avoir qu'une ligne par adresse IP, contenant un ou plusieurs noms d'hôtes.

Le premier nom est dit "*canonique*" (principal), les suivants sont des *alias*.



Une adresse IP n'est pas canonique, ça ne veut rien dire. Certains écrivent néanmoins plusieurs lignes avec la même adresse, mais le résultat n'est pas garanti.

Le fichier `/etc/hosts` est consulté pour la résolution de noms par les processus locaux, pour déterminer l'adresse IP associée à un nom d'hôte et vice versa.

Un navigateur web tournant sur une autre machine utilise son propre fichier hosts, pas le fichier hosts du serveur. Le DNS est une alternative qui permet de centraliser la résolution de noms sur des serveurs communs plutôt que de devoir gérer les fichiers hosts de chaque machine.

### rajouté par lagrenouille

Si tu n'as qu'une seule machine sur laquelle tournent à la fois le client et le serveur web, alors le fichier hosts suffit.

Le fichier hosts permet de faire correspondre facilement des adresses IP à des noms de domaine.

Le meilleur exemple mon réseau local où je me connecte sans avoir à taper l'adresse IP complète.

pour accéder en ssh à mon serveur, pas besoin de l'ip car il est dans le hosts

mon fichier hosts sur le serveur

```
$ cat /etc/hosts
127.0.0.1 localhost
127.0.1.1 lebug.home lebug
#127.0.0.1 localhost/nextcloud
127.0.0.1 funambule.org
#
# The following lines are desirable for IPv6 capable hosts
::1 localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

cette adresse est IP dans le fichier hosts de façon à ce que je n'ai qu'a taper:

```
<code> ssh lagrenouille@etmabule.org</code>
```

Lien sur le forum :

- [Alias locaux](#)

Merci aux participants de cette discussion pour avoir permis la rédaction de ce tuto. 😊

N'hésitez pas à y faire part de vos remarques, succès, améliorations ou échecs !

From:

<http://debian-facile.org/> - **Documentation - Wiki**

Permanent link:

<http://debian-facile.org/doc:systeme:hosts>



Last update: **30/04/2023 11:24**